Revgest: Augmenting Gestural Musical Instruments with Revealed Virtual Objects

Florent Berthaut, Cagan Arslan, Laurent Grisoni

Revgest: Augmenting Gestural Musical Instruments with Revealed Virtual Objects

Florent Berthaut
Univ. Lille, INRIA, CRIStAL, France
florent.berthaut@univ-lille1.fr

Cagan Arlsan
Univ. Lille, INRIA, CRIStAL, France
cagan.arslan@ed.univ-lille1.fr

Laurent Grisoni
Univ. Lille, INRIA, CRIStAL, France
laurent.grisoni@univ-lille1.fr

ABSTRACT

Gestural interfaces, which make use of physiological signals, hand / body postures or movements, have become widespread for musical expression. While they may increase the transparency and expressiveness of instruments, they may also result in limited agency, for musicians as well as for spectators. This problem becomes especially true when the implemented mappings between gesture and music are subtle or complex. These instruments may also restrict the appropriation possibilities of controls, by comparison to physical interfaces. Most existing solutions to these issues are based on distant and/or limited visual feedback (LEDs, small screens). Our approach is to augment the gestures themselves with revealed virtual objects. Our contributions are, first a novel approach of visual feedback that allow for additional expressiveness, second a software pipeline for pixel-level feedback and control that ensures tight coupling between sound and visuals, and third, a design space for extending gestural control using revealed interfaces. We also demonstrate and evaluate our approach with the augmentation of three existing gestural musical instruments.

Author Keywords

gestural musical instruments, revealed augmented reality, mixed-reality, revealed virtual objects

ACM Classification

H.5.5 [Information Interfaces and Presentation] Sound and Music Computing, H.5.2 [Information Interfaces and Presentation] User Interfaces

1. INTRODUCTION

Searching for new opportunities of musical expression, researchers and instrument designers have been investigating (among others) gestural interfaces.

From a hardware point of view, various signals from the body may be used, through devices that allow to use instrumented (or not) hand movements [13], finger movements and hand poses [5], or muscular activity [7]. Research on gestural instruments now provide knowledge about how to increase the transparency of Digital Musical Instruments through metaphors of physical world actions [8] or gestural sonic affordances [1]. They also create new opportunities for expression as they are more closely linked to the musician's body and as they remove some physical constraints of object-based instruments, e.g. on the amplitude of gestures.

In the context of computer-based instruments the lack, or limited use, of interaction with physical objects may also restrict feedback, both for the musician and the audience, and also restrict the appropriation [9] possibilities offered by the instrument. Visual feedback (with limited resolution) can be obtained by adding LEDs on the interfaces, for example using handheld devices as described by Hattwick et al. [10] or to glove interfaces such as the mi.mu gloves [15]. Richer visual feedback is usually displayed distant from the interface, for example on a screen on stage in front of the musician, as done for some of the T-stick performances [13]. However, as shown by Berthaut and Jones on control surfaces [3], the visual feedback designed by musicians, if relevant, often requires higher resolution as well as co-location with the gestures. It can further be used, for example, for providing information on context or exact values of sound parameters. Outside the musical field, Sodhi et al. [19] describe how 3D guides can help learn gestures

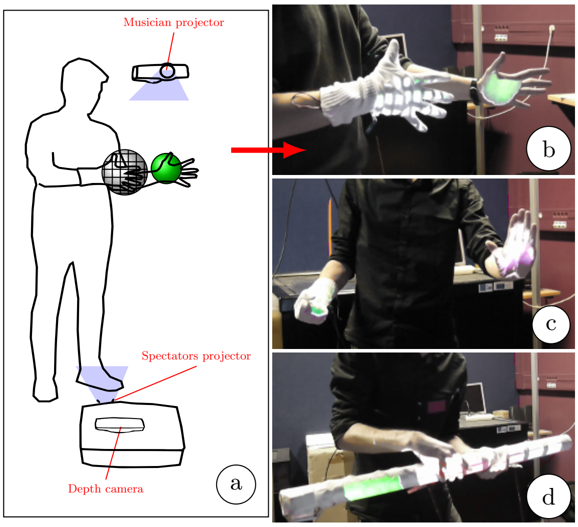

Figure 1: (a) Revgest in a setup with two projectors and one depth camera for public performance. The top projector is placed behind the musician and allows for feedback visible only to them. A virtual sphere is attached to the musician's right hand and provides feedback on finger movements sensed by a glove, another sphere in green controls a delay effect. (b) The resulting augmented gestural instrument. (c) Another glove based gestural instrument. (d) Augmentation of a handheld instrument.

NIME'17, May 15-19, 2017, Aalborg University Copenhagen, Denmark.
.

when projected on the hand to indicate directions. Feedback may also be important for the audience as shown by Berthaut et al. [2], as it may help spectators understand the causality link between gestures and the sound produced, especially when the exclusivity (e.g. multiple possible cause for a change in the sound) or consistency (e.g. a continuous gesture triggering discrete changes in the sound) criteria for agency are not respected by the instrument.

In this paper, we propose to revisit combination between digital musical instruments and physical objects in gestural interfaces, in the form of virtual objects that allow for additional visual feedback while preserving the focus on the gestures. These objects also extend the control and appropriation possibilities of gestural instruments.

Our contribution is three-fold: 1) We present a novel approach for extending gestural instruments using revealed virtual objects, allowing for additional feedback and expressiveness, while preserving their specificity. 2) We describe a software pipeline for feedback and control with tight coupling between sound and visuals, that we evaluate with measurements. 3) We provide a design space for extending gestural instruments with revealed virtual objects, that we demonstrate through the augmentation of existing instruments.

2. REVGEST

In this section we present our approach for extending gestural musical instruments using revealed augmented reality, together with a pipeline that allows for control and feedback with a tight coupling between sound and image, and a design space for revealed gestural controllers.

2.1 General approach

Our approach relies on two main principles.

First we propose to reintroduce objects in gestural instruments, in order to extend feedback and expression opportunities. This principle has already been explored in other fields of human-computer interaction and for generic gestural interfaces. In their work on Reality Based Interaction, Jacob et al. [11] analyse post-WIMP interfaces and suggest that they *draw strength by building on users' pre-existing knowledge of the everyday, non-digital world.* While gestural instruments tend to rely solely on what Jacob et al. name *Body Awareness & Skills*, bringing back virtual objects enables other categories of reality based interactions, such as *Naive physics*, in particular *the persistence of objects*, and *Environment Awareness & Skills*, in particular *skills to manipulate objects in their environment.* In the context of generic gestural interfaces, Rateau et al. [18] propose to create mid-air ephemeral and invisible screens, called mimetic interfaces, that can be used to facilitate control of distant displays. Fels et al. also rely on the metaphor of sculpting a virtual object [8] to add transparency to gestural control. The objects that we propose to add will first act as displays that can be placed anywhere in the physical space, attached to the musician's body, to handheld devices or statically placed in mid-air or intersecting objects, as shown in Figure 1.a. They will then serve as controls, offering metaphors such as touching, entering and exploring sounds and opening opportunities for appropriation, such as using different body parts or additional physical objects.

Second, we display these virtual objects, placed in the physical space, through the use of revealed augmented reality, rather than other augmented reality techniques. Revealed augmented reality (AR) is similar to slicing displays, which are used to explore layers of volumetric data, with the difference that the slicing is performed and appears in the physical space, not through a screen. Cassinelli et al. [6] use revealed augmented reality to annotate volumetric data. Martinez et al. [14] combine revealed augmented reality and optical combiners in order to project virtual content on top of physical objects. This principle is used again by Berthaut et al. [4] in the context of augmented musical performances. In their paper, the authors comment on the difference between revealing objects, i.e. revealing as a display, and augmenting gestures, i.e. revealing as a control. However, their system requires a mirror, few details are given on the implementation and opportunities of revealed augmented reality for gestural control are not explored. In our approach, virtual objects are directly revealed by, and thus visible on, the musician's body or a handheld device, as shown in Figure 1.b. This ensures that the focus for the musician and spectators remains on the gestures themselves, and not on the virtual objects. In the next section, we describe a pipeline for revealed AR that allows for what we call *pixel-level control and feedback*, ensuring a tight coupling between sounds and visuals. On one hand virtual objects can be placed at any 3D position, and their appearance precisely defined with 3D images pixel per pixel, so that the feedback resolution only depends on the projector resolution. They can also be dynamically modified with OpenSoundControl messages to reflect changes in the sound. On the other hand, revealing of objects can be detected and output at the pixel-level, meaning that a sound can be triggered even if a single pixel of an object appears.

2.2 Pipeline for pixel-level control and feedback

We propose a novel revealing pipeline that facilitates feedback through a variety of possible shapes and content. It also allows for the control of sound using the precise and fast output of position, extent and colour of the revealed part of the objects. Both feedback and control happen at the *pixel-level*, meaning that the resolution to display information and sense interaction is maximized.As depicted on Figure 2, the pipeline allows for defining scenes of virtual objects, which can be placed and revealed by body parts or objects in the physical space. The pipeline is available as part of the Revil software at `http://forge.lifl.fr/Revil`.

2.2.1 Sensing the physical space

In order to display the virtual objects, our pipeline first needs to sense the physical space, by which the objects will be revealed and in which they will appear. The first possibility is to scan the physical space using a depth camera, with either structured infrared light or time-of-flight technology. From the image, a 3D mesh is created and transformed to world coordinates. An example setup is shown in Figure 1.a. The second possibility is to track revealing physical objects and assign their transformation to 3D models with matching shapes. In the scenarios described in Section 3.1, we demonstrate both of these possibilities. Each physical element, depth camera and objects, is also assigned a unique identifier that can later be used to determine by which each virtual object was revealed.

2.2.2 Defining and controlling virtual objects

Virtual objects that are displayed by our pipeline are composed of a surface that encloses an inner volume. They are given a unique identifier that will be used during the revealing process. They can be statically placed with absolute coordinates, combined/transformed , and attached to tracked body parts or objects using OpenSoundControl messages sent to the software. It is possible to use different shapes, both primitives such as spheres, boxes, cylinders, but also 3D models and 3D paths, to define the surface of the virtual

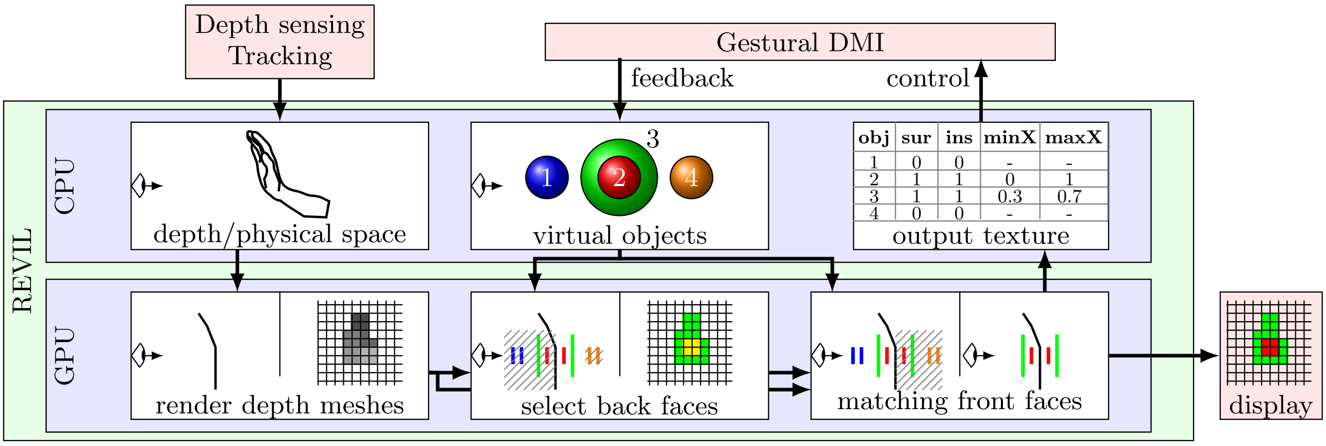

Figure 2: Software pipeline for pixel-level feedback and control. Here a hand reveals a red sphere inside a green sphere.

objects. These can be chosen to match the shape of physical elements they will be revealed with, or according to desired interaction with them. For example, revealing a sphere will result in a circular intersection of changing radius, that can be used to fade in a change in the sound, whereas a box will have the same cross-section at any position along its axes. The surface thickness can also be defined for example to ensure that it can be revealed separately from the inside.

The choice of content inside the objects depends on the type of feedback needed by the musician and audience. Color and texture gradients help perceive the revealed position (including depth) inside the virtual objects. Texture layers may be used to display volumetric data, such as 3D scans, or images that remain constant at all depths. These textures can be transformed, e.g. scaled, rotated or translated, to provide feedback on various parts of the instrument. Text content may be used to place labels in the physical space, either attached to the instrument or body, or at fixed positions. Finally, to facilitate appropriation [3], cuts of desktop windows can also be used as internal textures, allowing the musician to select parts of the existing graphical user interface of their software application and display them directly on their instrument / body.

Our pipeline handles multiple projectors, that can be placed at different positions in the physical space. For each virtual object, one can also select on which projector(s) it will be displayed. That enables displaying some objects only on projection seen from the musician's point of view, and other on the projection only seen from the audience, as depicted on Figure 3.g and 3.h.

2.2.3 *Revealing process*

The entire process of revealing runs on the graphics card GPU and consists of programs written in the OpenGL Shading Language (GLSL). As revealing is done per projected pixel, this process has to be repeated for each projector, each having its own position and orientation within the physical space and aspect ratio. It is composed of three passes. During the first one, the sensed physical space is rendered to a *slice* texture. Each pixel stores the distance to the projector and physical element id. For example in Figure 2, the hand that appears in the depth image is rendered to the *slice* texture.

During the second pass, the virtual scene, more specifically all objects selected for a projector, is rendered a first time to what we call the *select* texture. We select pixels of the back faces that are behind the *slice* texture. We store their identifier in the fourth color component of the pixel, transformed to a position in the 32 bits component value, so that a maximum of 32 objects can be revealed at once. We also store the result of a distance test between the *slice* texture pixel and the back face pixel, to know whether it is the surface, which has a defined thickness, or the inside that is being revealed. For example in Figure 2, the red and green objects have back faces behind the *slice* pixels and will be both present in the *select* texture.

During the third pass, we render the virtual scene again, but only the front faces. In the fragment shader, we keep pixels of objects when they are before the *slice* texture and the id of the objects have been stored at the same pixel in the *select* texture. This means that for this pixel, the back face of the object is behind the physical element and the front face in front of it, therefore the physical element intersects the virtual object. For example in Figure 2, pixels from the red and green objects appear the *select* texture, and current pixels are in front of the *slice* texture, so they are rendered.

For each rendered pixel, we use the absolute position of the *slice* pixel, computed from the distance and direction of the current projector, and the bounding box of the virtual object, to obtain the position ratio inside the bounding box, and display the correct pixel from the volumetric content. The rendered pixels are projected in the physical space on the elements that revealed the virtual objects. In our example, the intersection of the red and green objects with the hand will be projected back onto the hand.

2.2.4 *Output for control*

In the final pass of the GLSL program for each projector, as pixels are projected back into the physical space, we simultaneously retrieve which virtual objects are revealed and how, so that they can be sent to the instrument for additional control. Because this process is done at the pixel-level during display, the smallest change in what is displayed can be output for control, ensuring closely coupled auditory and visual feedback. Thus a single pixel of an object getting revealed can trigger a change in the music. This part of our pipeline makes heavy use of the *image load store* feature available in OpenGL core since version 4.4 [12]. This feature allows for writing data at arbitrary positions in textures and provides atomic operations to handle concurrent access between pixels, which on the GPU are processed in parallel.

In our case, an output texture is used with, as shown on Figure 2, one line per virtual object in the scene, and groups of columns for each physical element describing if/how the objects are revealed by the physical elements: is the surface revealed, is the inside revealed, what is the center of the revealed part, its extent, and the average color. The process in the fragment shader of the final pass is as follows :

For each pixel to be displayed, i.e. of a virtual object revealed in the physical space, we select the line in the output texture from the object identifier, and the group of columns matching the physical element id in the slice texture, so that a same object can be revealed by multiple identified elements, e.g. fingers. If the pixel of the *slice* texture is displayed either inside or on the surface (depending on the distance to the current pixel and on the value we stored in the *select* texture), we respectively set the the first or second columns using the max atomic operation. In order to output the center and extent of the revealed part together with the average pixel color, we retrieve the minimum and maximum on each axis using the min and max atomic operations. The values are normalized between 0 and 1, representing the ratio of position within the bounding box of the virtual object. The average colour is computed by accumulating the color components and counting the total number of pixel revealed for the object, with the add atomic operation. At the end of the rendering, the output texture is processed to retrieve the inside, surface, center, extent and average color for each virtual object. These values are then be sent to the gestural instruments with OpenSoundControl messages for extended control.

2.3 Design space for revealed gestures

In this section, we present a design space composed of four dimensions, that describes the opportunities for additional control and feedback opened by our approach for designers of gestural instruments. One must note that, for a given instrument, different values can be set over time, or multiple values can be combined for each of the dimensions. Examples for each of these are given in Figure 3.

2.3.1 Attachment

The *attachment* dimension pertains to how virtual objects can be placed in the physical space and their physical relation with the musician's body. It has three values :

attached to world (AW) : the object is placed in the physical space around the musician, at absolute coordinates. It can be static, for example to define volumes with different mappings presets on stage, or as 3D paths that act as continuous controls. It may also be moving, for example so that it displays automation on musical parameters through changes in positions of associated virtual objects.

attached to body (AB) : the object is attached to and follows the hand or other body parts of the musician. It can therefore be used as a permanent display when placed at the intersection with the body. It can also serve as an additional control when placed around it, for example allowing for bi-manual interaction, i.e. following one hand and revealed by the other.

attached to object (AO) : the object is attached to a handheld or static physical object. It can be mapped to its surface, acting as a traditional display, or placed around the physical object, which then serves as a spatial reference, allowing for above-the-surface interaction.

2.3.2 Control

The *control* dimension pertains to how virtual objects are used for additional control of the music, complementing the expression offered by the gestural instrument. It has three possible values :

no control (CN) : the object is only used for feedback.

discrete control (CD) : the object is used for discrete control, such as activating sequences or effects, triggering notes and so on. This corresponds to entering and leaving virtual objects, and going from revealing the surface to revealing the inside of objects. Surface may be used as an indication of distance to the inside, e.g. to know when one is about to enter the object and therefore trigger a change in the sound.

continuous control (CC) : the object is used for continuous control of musical parameters. The control may be a 3D position inside a virtual volume, a position along a path, a revealed color/texture. Example use cases include exploration of parameter spaces, playing through waveforms / sequences, exploration of audiovisual textures, and so on.

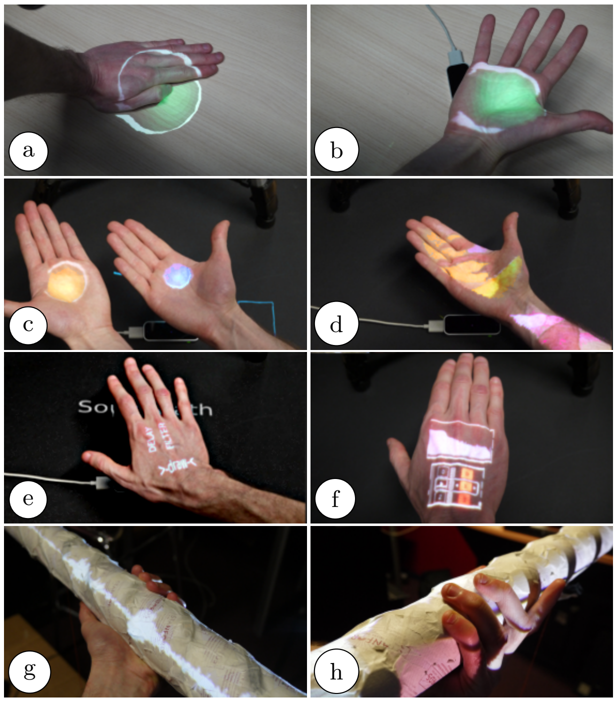

Figure 3: Illustration of the design space dimensions: (a) Attached to World (to the table), (b) Attached to Body (to the hand), (c) Discrete Control (activation of two sinewaves) , (d) Continuous Control (color components revealed in image layers control the volume of three sinewaves), (e) Mappings Feedback (pose labels), (f) Content Feedback (spectrum and position in pattern), Visibility different for musician (g) and spectators (h) on a handheld instrument.

2.3.3 Feedback

The *feedback* dimension describes the type of information that can be displayed. The three values correspond to the three categories described by Berthaut and Jones [3] after their study of appropriation by musicians :

mappings feedback (FM) : the object is used to identify the mappings between sensors and sound parameters, with text labels, textures or colours. The feedback can be static, e.g. for learning purposes, or dynamic to illustrate mapping selection or changes during performances.

parameters feedback (FP) : the object is used for feedback on musical parameters, in order to provide their context (curves, controlled waveform/timeline) or their exact values with dynamic labels.

content feedback (FC) : the object is used to provide the status of musical processes with for example VU meters, activation of sequences, position in a timeline with upcoming events, and so on.

2.3.4 Visibility

The *visibility* dimension pertains to who sees the augmentations. It has three possible values:
musician only (VM): the virtual objects is revealed only, and therefore seen only, on the musician's side. It can be used for feedback needed by the musician but too small / complex to add anything to the spectators experience.
similar for musician and audience (VS) : the virtual object is revealed with the same aspect for both the musician and spectators.
different for musician and audience (VD) : the virtual object has a different aspect when seen from the musician's side than from the spectators' side. Feedback with different levels of detail can therefore be provided, for example precise value of parameters for the musicians and the same value represented by a color scale for the audience, or full waveform for the musician and a VU meter for the audience. This can be achieved, as shown in Figure 1.a by using projectors around the musician.

3. EVALUATION

In this section, we first evaluate our design space using the augmentation of three existing gestural instruments. We then evaluate our implementation and discuss its limitations in the case of off-the-shelf hardware.

3.1 Design space evaluation

In this section, we present the implementation of simplified versions of three existing gestural instruments, and demonstrate how they can be augmented using the design space described in 2.3. For each instrument, we present both a feedback only version, and a version with extended control.

3.1.1 Xth sense

The first instrument takes inspiration from the xth-sense [7] instrument, which relies on MMG and EMG signals to control the sound. We recreate the system using the same glove described above. An additional pressure sensor on the palm activated when the hand is squeezed, evaluates the strength of the grasp. In our version, four white noise generators are controlled by the movement of the fingers. Movement speed is mapped to the volume and bending is mapped to a different voltage-controlled filter for each finger. Grasp strength is used to control a filter on the overall sound. Finally, the average finger extension is used to control the wet/dry on a reverb placed before the main filter, allowing one to trigger long reverberation by suddenly opening their hand. In the feedback only version, a single object is attached to the hand, visible by both the musician and the audience. It is used to visualize the grasp strength, mainly to increase the agency of the audience, by mapping the grasp strength to the color saturation of the virtual object. In addition, the scale of the object reflects the output loudness of the instrument. This increases the consistency criteria for agency [2] when the sound is less dependent on the gestures, such as when the reverb is activated . This feedback only version can be classified in our design space as *VS, FP/FC, CN, AB*. The extended control version, shown on Figure 1.b, adds three objects around the hand and attached to it. By intersecting these objects, the musician activates three separate delay effects. This version creates opportunities for appropriation as the objects can be revealed by any physical elements and move together with the hand. This version can be classified as *VS, FP/FC, CD, AB*.

3.1.2 Soundgrasp

The second instrument is an adaptation of the musical glove presented by Mitchell and Heap [16]. We created a data glove to reproduce the SoundGrasp system. Four flex sensors positioned on the proximal interphalangeal joints except that of the thumb, measure the opening of the hand. The sensor outputs are connected to analogue inputs of a x-OSC board which sends OSC packets to the server. Our simplified version allows one to record their voice by opening the hand, and to loop the recorded phrase by closing it. Two effects can be applied by extending one or two fingers. For the feedback only version of the augmented instrument multiple objects are attached to the hand and display labels and colors on the instrument status (recording / playing) in order to guide the musician's interaction. This version can therefore be placed in the design space with the following values *VS, FM/FP, CN, AB*. The extended control, shown on Figure 1.c, adds the possibility of creating a 3D sound-path placed in mid-air when the loop is played for the first time. This path is revealed by the musician and visible for both him and the audience. The musician may then activate and deactivate the playing loop by entering and leaving the path. The center and extent the section revealed in the path then controls the length and starting point in the loop. This version is classified the design space with the following values *VS, FM/FP, CD/CC, AB/AW*.

3.1.3 T-stick

The third instrument draws inspiration from the t-stick developped by Malloch et al. [13]. It consists in a tube equipped with various sensors. Sound parameters can then be controlled with the movements of the tube and by pressing, sliding, tapping the tube. In our version, 10 pressure sensors are placed along the length of the tube. They control the volume of 10 granular synthesizers which play ten positions along the same sound. The speed of the tube controls a global volume. In the feedback only version an object displaying the waveform is attached to the tube, so that the sensors are aligned with their positions. It is visible only from the musician's point of view. From the audience side, only the pressure is amplified, by changing the color saturation of objects placed on the tube aligned with the sensors. These two sides can be seen on Figure 3.g and 3.h. This version can be classified as *VD, FP, CN, AO* In the extended control version, shown on Figure 1.d, three large zones are defined in the physical space. Each zone is associated to an effect, activated when the musician is inside the zone. In addition to the visual feedback given to both musician and audience on which effects are active, these virtual objects open expression opportunities, such as combining effects or playing with the borders of zones for glitches in their activation. This version can be classified as *VD, FP, CD, AO/AW*.

3.2 Technical evaluation

We first evaluate the latency of our system. To do so, we measured the time interval between when an OpenSoundControl message is received to make a virtual object visible by changing its position, and just before the output revealing message is sent. The operation sequence is therefore as follows : access the virtual object, change its the model matrix, render the scene, process the output texture to get the number of points. Results of the measurements for varying number of revealed pixels and for two different systems are shown in Figure 4. On both these systems the rendering is synchronised with the vertical retrace of the projectors, which happens at 60Hz, therefore fixing our lower latency limit to 16.6ms. Limitations in the revealing speed originate from both the refresh rate of the projector and the framerate of the depth camera used. The Asus Xtion that we use for our implementation provides a resolution of 320x240 pixels at 60fps. At this framerate, the latency introduced by the depth camera can however be quite noticeable during fast movements, for example leaving "shadows" of the revealed vitual objects projected on surfaces behind the body. Latency issues can be solved in part by the development of high speed tracking and projection devices, such as the ones presented by Narita et al. [17]. Rigid body 3D tracking systems also provide higher framerates, but do not capture the entire physical scene.

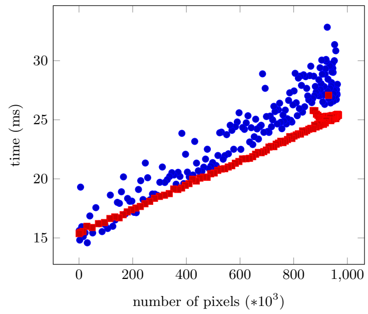

Figure 4: Time needed to move a sphere, reveal (display) it and process the output for control, depending on the number of revealed pixels. In red with a NVIDIA Quadro M2000 GPU and an Intel Xeon CPU (2.67GHz) processor, in blue with an integrated Intel HD Graphics 520 and an Intel i7

Two other limitations of our approach are the occlusion issues which come from the use of projectors, the musician's own body may in fact hide parts of the projection during, and the decrease in resolution of the revealed objects when the distance to the projectors increases. Solutions include increasing the number of projectors, in order to cover multiple angles and distances of projection, or using flexible wearable displays.

3.3 Impact evaluation perspectives

Further evaluation of the impact of Revgest can be done on two main aspects. First the feedback provided for the audience might be evaluated with regard to the perceived control that the musician has over the sound. This perceived control, or causality link, might be reduced as explained in [2] because of small/subtle/unfamiliar gestures, complex mappings and automated musical processes. Following the study run by Berthaut et al., videos of gestural instruments with and without revealed virtual objects could be shown to spectators to evaluate their perception of the liveness of the instruments. On the musician's side, we plan to evaluate the impact of additional feedback with a long term study by observing differences between practices and gestures without revgest, as done in [3], and those after a long period of appropriation and use of revgest.

4. CONCLUSION

In this paper, we presented Revgest, a novel approach for augmenting gestural musical instruments with revealed virtual objects. We contributed a software pipeline for revealing virtual objects that allows for pixel-level feedback and control, and a design space that describes the opportunities opened by Revgest. Future work will focus on the technical limitations described in section 3.2, by investigating alternative tracking and projection technologies.

5. REFERENCES

[1] A. Altavilla, B. Caramiaux, and A. Tanaka. Towards gestural sonic affordances. In *Proceedings of NIME*, pages 61–64, May 2013.

[2] F. Berthaut, D. Coyle, J. Moore, and H. Limerick. Liveness Through the Lens of Agency and Causality. In *Proceedings of NIME*, Baton Rouge, United States, May 2015.

[3] F. Berthaut and A. Jones. Controllar: Appropriation of visual feedback on control surfaces. In *Proceedings of ACM ISS*, pages 271–277, 2016.

[4] F. Berthaut, D. Martinez Plasencia, M. Hachet, and S. Subramanian. Reflets: Combining and Revealing Spaces for Musical Performances. In *Proceedings of NIME*, May 2015.

[5] D. Brown, N. Renney, A. Stark, C. Nash, and T. Mitchell. Leimu: Gloveless music interaction using a wrist mounted leap motion. In *Proceedings of NIME*, 2016.

[6] A. Cassinelli and M. Ishikawa. Volume slicing display. In *ACM SIGGRAPH ASIA 2009 Art Gallery; Emerging Technologies*, pages 88–88, 2009.

[7] M. Donnarumma, B. Caramiaux, A. Tanaka, et al. Muscular interactions combining emg and mmg sensing for musical practice. In *Proceedings of NIME*. KAIST, 2013.

[8] S. Fels, A. Gadd, and A. Mulder. Mapping transparency through metaphor: towards more expressive musical instruments. *Organised Sound*, 7(2):109–126, 2002.

[9] M. Gurevich, P. Stapleton, and A. Marquez-Borbon. Style and constraint in electronic musical instruments. In *Proceedings of NIME*, pages 106–111, 2010.

[10] I. Hattwick and M. M. Wanderley. Interactive lighting in the pearl: Considerations and implementation. In *Proceedings of NIME*, pages 201–204, 2015.

[11] R. J. Jacob, A. Girouard, L. M. Hirshfield, M. S. Horn, O. Shaer, E. T. Solovey, and J. Zigelbaum. Reality-based interaction: A framework for post-wimp interfaces. In *Proceedings of ACM CHI*, pages 201–210, 2008.

[12] Khronos-group. Opengl image load store. `https://www.khronos.org/opengl/wiki/Image_Load_Store`, 2017. [Online; accessed 30-December-2016].

[13] J. Malloch and M. M. Wanderley. The t-stick: From musical interface to musical instrument. In *Proceedings of NIME*, pages 66–70, 2007.

[14] D. Martinez Plasencia, F. Berthaut, A. Karnik, and S. Subramanian. Through the combining glass. In *Proceedings of ACM UIST*, pages 341–350, 2014.

[15] mi.mi. Mi.mu gloves. `http://mimugloves.com`, 2017. [Online; accessed 30-December-2016].

[16] T. Mitchell and I. Heap. Soundgrasp: A gestural interface for the performance of live music.

[17] G. Narita, Y. Watanabe, and M. Ishikawa. Dynamic projection mapping onto a deformable object with occlusion based on high-speed tracking of dot marker array. In *Proceedings of ACM VRST*, 2015.

[18] H. Rateau, L. Grisoni, and B. De Araujo. Mimetic Interaction Spaces : Controlling Distant Displays in Pervasive Environments. In *Proceedings of IUI*, Haifa, Israel, Feb. 2014.

[19] R. Sodhi, H. Benko, and A. Wilson. Lightguide: Projected visualizations for hand movement guidance. In *Proceedings of ACM CHI*, pages 179–188, 2012.